

教师科研业绩量生成和管理系统

—系统管理

院系名称：计算机学院

专业:软件工程

组员：201608040221王宏飞

201608040118庞棣

201608040135蔡明

指导教师： 张西广

2018 年 11 月20

摘要

本设计按照管理信息系统的开发方法,通过系统分析、系统设计、系统实施等一系列开发过程.以中原工学院为应用背景,开发一个有实用价值的教师科研业绩量生成和管理系统.实现对教师科研业绩量生成和管理,以提供及时,准确的信息服务并减轻管理人员制作报表,统计分析的负担,从而摆脱学院长期以来通过人工对教师科研业绩量进行管理的落后面貌.

该系统能对教师科研业绩量实现较全面的管理、提供数据录入、多条件查询、统计、报表、输出等日常管理所必需的功能,并在系统的易使用性上作了较大的努力.

该系统采用Javaee作为系统的开发语言,运行的环境为windows10操作系统,后天的数据库管理系统选用Mysql.

关键词：Javaee Mysql 教师科研业绩量生成和管理系统

目 录

//自动生成

第1章 引 言

## 1.1 背景

为了进一步深化我校人事制度改革，合理配置学校人力资源，加快推进岗位设置管理工作，全面落实岗位聘用制，进一步加强学科队伍建设，优化教职工队伍结构，改革人事管理及薪酬分配制度，激发教师工作活力，更好完成学校“十三五”的目标和任务，实现我校人事管理的科学化、规范化和制度化。我校于2017年进行了新一轮岗位聘用。根据学校文件要求，学校也出台了中原工学院业绩量化办法，目前教学业绩量管理还好，多年计算方法没多大变化，而科研业绩量的核算是一个新命题，以前没有经验积累，所以第一年的核算很是费时费力。因此，一套合理的、正确的、完整的教师科研业绩量生产和管理系统是必要的，也是具有现实意义的。

## 1.2 现状及发展趋势

目前国内外各高校均已拥有独立的校园网、运行相对稳定，但我国大部分高职院校仍采用半手工甚至手工方式来处理科研业绩管理工作，办公自动化还基本没有实现，也无法切实有效地开展科研业绩相关工作。早期使用的科研业绩管理系统基本是单机版，各系统之间的数据无法实现共享，几个系统分别采用的数据库也是相互独立的，从而给系统的使用和维护都带来了诸多不便。目前，各高职院校教师科研信息的维护工作(如信息发布、信息更新、信息统计、信息查询等)的任务十分繁重，单纯靠“闭门造车”式的管理方式显然已不合时宜。目前虽然大部分高职院校根据自身特点制定了一套教师科研业绩评价办法和标准，但并无适合的系统支撑，这必然引起校内评价实施过程中的诸多问题，无法保证科研业绩评价的公平原则。

## 1.3 研究内容

## 教师科研业绩量生成和管理系统是通过一套科学合理的体系,把教师在一定时期内的个人信息及个人获奖情况进行统计和归纳,并且给出分析与评测的结果。笔者通过对学校现行教师科研业绩评价体系进行分析，在充分利用数据库技术、j2ee技术和编程技术的基础上，构建了一套适用于我校实际的教师科研业绩管理系统，实现对教师科研业绩的过程化管理、科学评价和优化管理。系统与学校网现有的“教数据库”进行数据的无缝链接，一旦经登录认证过即可登录该系统。

## 1.4 课题意义和目的

进一步深化我校人事制度改革，合理配置学校人力资源，加快推进岗位设置管理工作，全面落实岗位聘用制，进一步加强学科队伍建设，优化教职工队伍结构，改革人事管理及薪酬分配制度，激发教师工作活力，更好完成学校“十三五”的目标和任务，实现我校人事管理的科学化、规范化和制度化。

## 1.5 相关技术

该系统文档描述中所用到的相关技术及其术语说明如表1-1所示：

表格 1-1 系统术语表

|  |  |
| --- | --- |
| 术语 | 说明 |
| 用例图 | 指由参与者（Actor）、[用例](http://baike.baidu.com/view/706238.htm" \t "_blank)（Use Case）以及它们之间的关系构成的，用于描述系统功能的静态视图。 |
| 时序图 | 描述对象之间发送消息的时间顺序显示多个对象之间的动态协作，是一种UML交互图.。 |
| 类图 | 显示了模型的静态结构，特别是模型中存在的类、类的内部结构以及它们与其他类的关系等。类图不显示暂时性信息。 |
| J2EE | J2EE是一套全然不同于传统应用开发的技术架构，包含许多组件，主要可简化和规范应用系统的部署与开发，进而提高可移植性、安全性与再利用价值。 |
| Springboot框架 | 用来简化新Spring应用的初始搭建以及开发过程。该框架使用了特定的方式来进行配置，从而使开发人员不再需要定义样板化的配置。 |
| Vue.js | Vue.js是一个构建数据驱动的 web 界面的渐进式框架。Vue.js 的目标是通过尽可能简单的 API 实现响应的数据绑定和组合的视图组件。它不仅易于上手，还便于与第三方库或既有项目整合。 |
| MYSQL | MYSQL是一个关系型数据库管理系统。 |
| JDBC | JDBC（Java Data Base Connectivity,java数据库连接）是一种用于执行SQL语句的Java API，可以为多种关系数据库提供统一访问，它由一组用Java语言编写的类和接口组成。 |
| JSP | JSP技术使用Java编程语言编写类XML的tags和Scriptlets，来封装产生动态网页的处理逻辑。网页还能通过tags和Scriptlets访问存在于服务端的资源的应用逻辑。JSP将网页逻辑与网页设计和显示分离，支持可重用的基于组件的设计，使基于Web的应用程序的开发变得迅速和容易。 |
| Spring | Spring是一个开源框架，Spring使用基本的JavaBean来完成以前只可能由EJB完成的事情。然而，Spring的用途不仅限于服务器端的开发。从简单性、可测试性和松耦合的角度而言，任何Java应用都可以从Spring中受益。简单来说，Spring是一个轻量级的控制反转（IoC）和面向切面（AOP）的容器框架。 |

第2章 需求分析

教师科研业绩量生成和管理系统包含五个子模块，即：系统管理、数据录入、数据统计、人工审核、帮助信息。本章节将详细讲述系统管理的需求分析，采用面向对象的分析方法，使用UML建立系统用例模型，创建系统功能模型。

## 2.1 总体需求

教师科研业绩量生成和管理系统系统的用户分为两类：系统管理员、教师用户。

**1、系统管理员**  (1)个人中心:维护自己的资料、修改个人密码和注销登录

(2)数据管理:修改教师科研项目的数据,人工审核数据

(3)用户操作:对教师用户的增删改查

**2、教师用户**

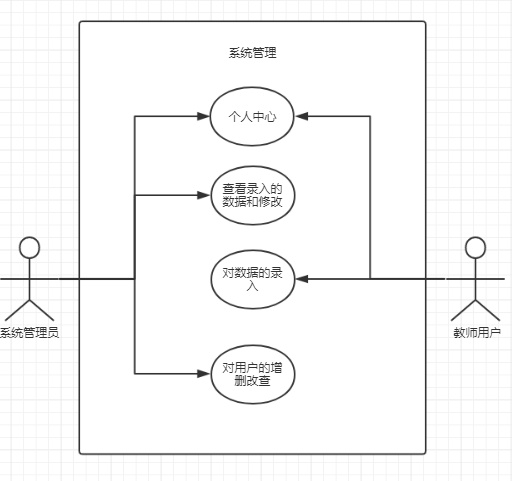
(1)个人中心:维护自己的资料、修改个人密码和注销登录

(2)录入个人科研项目数据和申请修改个人数据

(3)查看个人数据

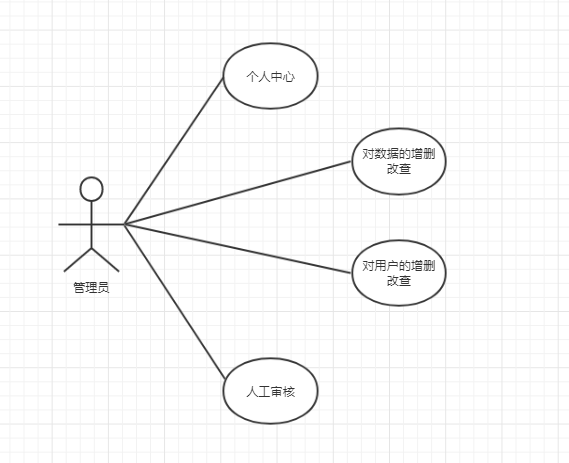
## 2.2 功能需求

对于系统管理的功能需求用例描述如下图所示



### 2.2.1 系统角色

系统管理员的功能有：个人信息修改,人工审核数据,对项目数据的增删改查,对用户的增删改查。如图所示:



### 2.2.2 需求描述

1. **登录**

关联角色：系统管理员、教师用户

需求编号：CRX-001

需求描述：系统数据库中存在的用户，无论什么角色，如系统管理员、教师用户，都可以登录进入该系统。

备注说明：该系统数据库中必须存在用户的登录账号和密码，并且用户的状态是“启用”时，才可进行登录操作。

1. **个人中心**

关联角色：系统管理员,教师用户

需求编号：CRX-002

需求描述：管理员登录后，可以执行查看自己的个人信息，修改密码，注销登录等操作。

备注说明：登录后，才可进行相关的操作。

1. **系统管理**

关联角色：系统管理员

需求编号：CRX-003

需求描述：管理员登录后，可以进行用户及用户数据修改操作,管理员通过查询用户姓名来修改和删除数据

备注说明：登录后，才可进行相关的操作。

1. **科研项目数据录入及查看**

关联角色：用户

需求编号：User-001

需求描述：用户登录后,可以进行用户自己本身的数据操作

备注说明：登录后，才可进行相关的操作。

## 2.3非功能需求

### 2.3.1 性能要求

1、系统支持多用户并发操作。

2、系统处理操作响应时间不超过10秒。

### 2.3.2 可扩展性

系统当前使用的人数不多且功能简单，对于后期使用人数的增多以及功能的修改和增删，都能够做出相应的扩展。

### 2.3.3 可用性

保证正常情况下系统可用，可允许系统每月停机一段时间，来进行相关的系统维护。维护时间可安排在恰当的非工作时间进行。

### 2.3.4 安全性

1. **操作跟踪**

针对系统运行出现的异常，跟踪调查出现异常的情况，有针对性的解决问题。

系统日志，便于查看系统的运行情况。

操作日志, 提供用户在系统中增加、修改系统数据信息时日志记录，用于跟踪用户的相关操作。

1. **访问控制**

页面不可以直接访问，防止他人对页面进行篡改。页面访问通过连接动作驱动有效防止用户通过地址栏输入地址而对信息非法访问作了权限检查相关操作。

### 2.3.5 易用性

系统简单、易用，具有清晰的导航功能，使用户能快速找到自己想要执行的操作页面。同时，也能保证用户通阅读用户手册，可以使用此系统。

### 2.3.6 开发环境

**1. 软件约束** 教师科研业绩生成和管理系统采用Eeclipse和MySQL为工具进行开发。

开发及运行的软件环境为：win10;

开发系统平台：Eclipse 4.9 + Tomcat8.5.29 + JDK1.8+Maven3.5.3；

浏览器：Internet Explorer 8.0或者以上版本、Googole、Firefox等；

数据库：MySQL;  
开发框架：Vue.js,springboot,j2ee框架。

1. **硬件约束**

操作系统：32/64位的 Windows 10系统

## 2.4 用例规约描述

本章节主要根据《软件需求规格说明书》，来进行的用例建模，并对用例模型进行具体描述。用例规约是面向对象分析和设计的重要步骤。

### 2.4.1 用例目的

用例是从外部角度描述用户和系统之间相互作用的行为，表达系统应该做什么。本文档通过用例规约，来进一步描述该系统需求，它是系统设计的基础，也是测试用例编写的重要依据。

### 2.4.2 用例描述

1. **管理员登录**

管理员登录系统后台的用例规约如表1-1所示:

表1-1 管理员登录用例规约

|  |  |
| --- | --- |
| 用例名称 | 管理员登录 |
| 用例ID | Zut\_Role\_001 |
| 用例说明 | 管理员登录系统 |
| 执行者 | 管理员 |
| 前置条件 | 无 |
| 主成功场景 | 1. 用户访问系统 2. 系统显示登录页面 3. 用户输入用户名和密码，选择登录身份，然后单击“登录”按钮 4. 系统验证用户名和密码、身份及用户的状态 5. 系统显示登录成功后的个人资料页面 |
| 扩展1 | (4a)用户名和密码为空、身份未选  (4a1) 系统弹出提示框“用户名或密码错误”，显示当前的登录页。  (4a2) 返回主成功场景步骤(3) |
| 扩展2 | (4b) 用户名或密码或身份填写错误或用户被禁用  (4b1) 系统弹出提示框“用户名或密码错误”，显示当前的登录页  (4b2) 返回主成功场景步骤(3) |
| 后置条件： | 登录成功，可以进行权限范围内的其他操作。 |
| 备注说明： | 只有系统数据库中存在的用户，并且用户状态为“启用”时才可按身份登录该系统。 |

1. **管理员修改用户数据**

管理员登录系统后台进行修改用户操作的用例规约如表1-2所示：

表1-2管理员修改用户用例规约

|  |  |
| --- | --- |
| 用例名称 | 管理员修改用户 |
| 用例ID | Zut\_Role\_001 |
| 用例说明 | 管理员操作系统管理模块的用户信息---修改用户 |
| 执行者 | 管理员 |
| 前置条件 | 登录成功 |
| 主成功场景 | 1. 管理员进入系统后台主页 2. 点击导航里的修改数据 3. 输入用户名称,进入修改页面 4. 输入修改信息并确定,”修改”对话框关闭,提示保存成功 5. 系统将信息更新到数据库中的用户表里，并返回到用户列表页中显示 |
| 扩展 | (5a)填写用户信息格式不对  (5a1) 系统无法提交信息更新数据库的用户表，“添加”对话框未关闭。  (5a2) 返回主成功场景步骤(4) |
| 后置条件： | 修改用户数据成功，可以进行权限范围内的其他操作。 |
| 备注说明： | 只有输入的信息格式正确，操作才可顺利进行 |

1. **管理员删除用户**

管理员登录系统后台进行删除用户操作的用例规约如表1-3所示：

表1-3管理员添加用户用例规约

|  |  |
| --- | --- |
| 用例名称 | 管理员添加用户 |
| 用例ID | Zut\_Role\_001 |
| 用例说明 | 管理员操作系统管理模块的用户信息---删除用户 |
| 执行者 | 管理员 |
| 前置条件 | 登录成功 |
| 主成功场景 | 1. 管理员进入系统后台主页 2. 选择后台主页左侧的菜单导航栏的“系统管理”，点击“系统管理”，展开其子菜单，点击“用户信息”，进入用户信息界面 3. 点击“用户信息”界面的“删除”按钮，弹出一个查找用户信息的对话框 4. 输入需要删除的用户的信息并确定，“删除”对话框关闭，提示保存成功 5. 系统将删除的信息从数据库中删除 |
| 扩展 | (5a)填写用户信息格式不对或者为空  (5a1) 系统无法提交信息更新数据库的用户表，“删除”对话框未关闭  (5a2) 返回主成功场景步骤(4) |
| 后置条件： | 删除用户成功，可以进行权限范围内的其他操作。 |
| 备注说明： | 只有输入的信息格式正确，操作才可顺利进行 |

1. **管理员修改数据**

管理员登录系统后台进行修改用户操作的用例规约如表1-4所示：

表1-4 管理员修改用户用例规约

|  |  |
| --- | --- |
| 用例名称 | 管理员修改用户 |
| 用例ID | Zut\_Role\_001 |
| 用例说明 | 管理员操作系统管理模块的用户信息---修改数据 |
| 执行者 | 管理员 |
| 前置条件 | 登录成功 |
| 主成功场景 | 1. 管理员进入系统后台主页 2. 选择后台主页左侧的菜单导航栏的“系统管理”，点击“系统管理”，展开其子菜单，点击“用户信息”，进入用户信息界面 3. 选择“用户项目数据”界面中用户项目信息列表中的某一条记录，点击“修改”按钮，弹出一个被选中用户信息的修改对话框 4. 输入选中用户的修改信息并确定，“修改”对话框关闭，提示保存成 5. 系统将信息更新到数据库中的用户表里，并返回到用户列表页中显示 |
| 扩展 | (5a)填写用户修改信息格式不对  (5a1) 系统无法提交信息更新数据库的用户表，“添加”对话框未关闭。  (5a2) 返回主成功场景步骤(4) |
| 后置条件： | 修改数据成功，可以进行权限范围内的其他操作。 |
| 备注说明： | 只有输入的信息格式正确，操作才可顺利进行 |

1. **管理员查询用户**

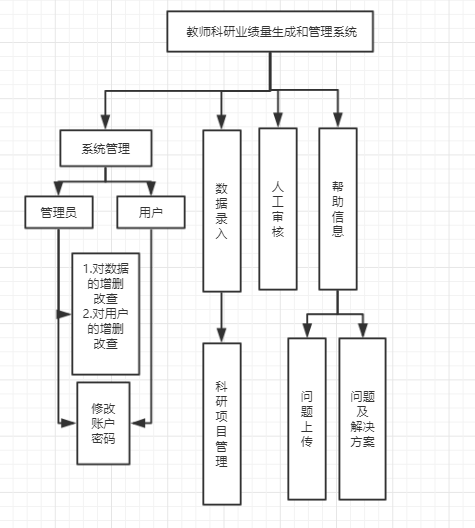
管理员登录系统后台进行查询用户操作的用例规约如表1-5所示：

表1-5 管理员查询用户用例规约

|  |  |
| --- | --- |
| 用例名称 | 管理员查询用户 |
| 用例ID | Zut\_Role\_001 |
| 用例说明 | 管理员操作系统管理模块的用户信息---查询用户 |
| 执行者 | 管理员 |
| 前置条件 | 登录成功 |
| 主成功场景 | 1. 管理员进入系统后台主页 2. 选择后台主页左侧的菜单导航栏的“系统管理”，点击“系统管理”，展开其子菜单，点击“用户信息”，进入用户信息界面 3. 在查询框中输入要查询用户的用户名，并点击“查询”按钮 4. 系统将查询框中的信息与数据库中用户表里的用户名进行模糊匹配，并将得到的结果返回到用户列表页中显示 |
| 扩展 |  |
| 后置条件： | 查询用户成功，可以进行权限范围内的其他操作。 |
| 备注说明： | 只有输入用户名这个查询条件，操作才可顺利进行 |

第3章 系统分析

如下图所示,系统分为系统管理、数据录入、数据统计、人工审核、帮助信息五大模块.



**系统管理:**

分为两个实体用户:管理员和教师用户,并赋予实体应有的权限

**数据录入:**

支持校内数据的自动导入和教师手动录入功能

**数据统计:**

实现对单个教师科研业绩核算、部门科研业绩分汇总,支持管理权限下教师个人、部门管理人员、学院院领导的科研业绩在线查询

**人工审核:**

实现对一系列教师科研成果（如科研项目、教改研究、大赛获奖、教材、专著、论文、专利、软件著作权）的过程话审核,实时进行网页提醒.

**帮助信息:**

提供窗口来实现对本系统的问题汇聚以及提供常见问题解决方案

## 3.1 静态模型

### 3.1.1 实体类（根据变成人员的编码来提供———未完成）

1. 有属性
2. 有操作
3. 与其他类有关联

类描述了一组对象的公共属性和行为，类为对象规定了它们的属性和所能提供的操作。下面是该系统中系统管理员权限内各个类之间的关联模型图。这些类包括:

管理员、教师用户、权限

### 3.1.2 边界类（根据变成人员的编码来提供———未完成） 系统的用户通过边界类与系统打交道，用户向边界类发起访问，边界类把用户的申请提交给控制类，控制类最终访问实体类。一般在进行设计时可能一个用例对应一个边界类，也可能是一个用户对应一个边界类，有时一个边界类可能还要调用另一个边界类才能完成用例。

### 3.1.3 控制类

边界类通过控制类访问实体类，这是三层架构设计体系结构，本系统一个实体类基本有一个控制类做管理。系统主要的控制类如表所示：

## 3.2 时序图的分析 时序图主要描述了不同对象之间发送消息的时间顺序显示多个对象之间的动态协作。本系统基本上对每个用例建立了一个时序图，建立时序图是为了进一步描述用例。 本项目的体系结构采用三层架构规范。其中: 表示层，用于与用户进行交互并显示结果，在时序图中由边界类来实现这一层。 控制层，主要完成三项任务，一是调用实体类，二是业务逻辑验证，三是决定将合适的视图组件返回给用户。 模型层，主要由实体类组成，包括一些基本的逻辑。

1. **登录时序图**
2. **修改密码时序图**
3. **用户信息管理时序图**
4. **管理员信息管理时序图**
5. **人工审核管理时序图**
6. **数据修改管理时序图**
7. **用户删减管理时序图**

第4章 系统设计

## 4.1 架构设计

本系统采用基于Web服务的B/S结构（即浏览器和服务器结构）的架构。在B/S结构下，用户操作界面是通过Web页面来展示实现，极少部分的事务逻辑需要在前端（Browser）实现，但是主要事务逻辑操作在服务器端实现，形成所谓三层Three-tier结构。这样就大大简化了客户端电脑载荷，减轻了系统维护与升级的成本和工作量，降低了用户的总体成本。B/S架构管理软件更方便、速度快、效果优。

系统使用基于springboot模式的三层架构，分为View表示层、Controller控制层、JavaBean数据访问层以及数据库，如图4-1所示：



图4-1 springboot模式的三层架构图

## 类设计

**1.用户信息User类**

用户信息类包含教师用户登录名u\_ teacherId、密码u\_password、真实名称u\_name、年龄u\_age、联系电话u\_phone、部门u\_post，用户类的方法有登录login()、修改个人信息alterPerson()、录入数据addData()、查询数据findData()和修改数据update()、问题查询question()，设计用户类图如图4-2.1所示：

|  |
| --- |
| User |
| -u\_teacherId  -u\_password  -u\_name  -u\_age  -u\_phone  -u\_post |
| +login()  +alterPerson()  +addData()  +findData()  +update()  +createTable()  +question |

图4-2.1教师用户类

**2.权限Auth类**

权限类包含属性权限a\_id、权限名称a\_name、权限描述a\_explain，权限类的方法有登录权限login()、修改个人信息alterPerson()、录入数据a\_addData()、查询数据a\_findData()、修改数据r\_alterData()、查询用户r\_findUser()、删除用户r\_delteUser()、创建绩点表a\_createTable()、人工审核r\_review()、问题描述question()，设计权限类图如图4-2.2所示：

|  |
| --- |
| **Auth** |
| -a\_id  -a\_name  -a\_explain |
| +login()  +alterPerson()  +addData()  +findData()  +alterData()  +r\_findUser()  +r\_delteUser()  +createTable()  +r\_review()  +question |

图4-2.2权限类

**3.管理员Role类**

管理类包含管理登录账户r\_id、管理登录密码r\_password、管理名称r\_name，管理类的方法有登录login()、修改个人信息alterPerson()、查询用户r\_findUser()、删除用户r\_deleteUser()、查询数据findData()、修改数据alterData()、人工审核r\_review()、问题描述question()，设计角色类图如图4-2-3所示：

|  |
| --- |
| **Role** |
| -r\_id  -r\_password  -r\_name  -r\_age  -r\_phone  -r\_post |
| +login()  +alterPerson()  +r\_findUser()  +r\_delteUser()  +findData()  +alterData()  +r\_review()  +question() |

图4-2-3管理员Role类

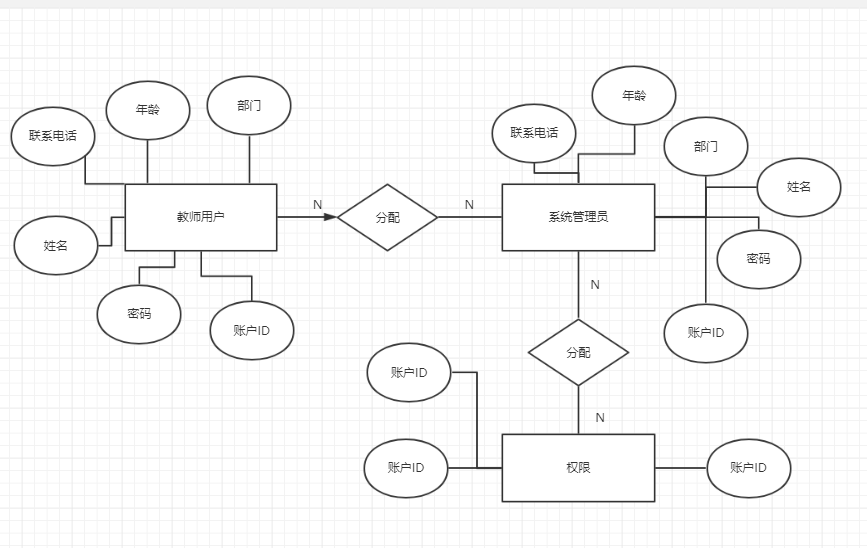
**4.项目绩点计算Count类**

绩点计算类包含项目编号c\_id、项目名称c\_name、项目描述c\_explain、绩点项目分类名称c\_projectName、总绩点c\_count，管理类的方法有Count类，设计角色类图如图4-2-4所示：

|  |
| --- |
| Count |
| -c\_id  -c\_name  -c\_explain  -c\_projectName  -c\_count |
| +Count |

## 数据库表设计

1.**管理用户**



4.3.2 物理模型

**1.教师用户信息表** 教师用户信息表(User\_tb)用户存放用户的基本信息,如下图4-3-2.1所示

|  |  |  |
| --- | --- | --- |
| 字段名 | 字段类型 | 字段描述及说明 |
| u\_teacherId | Int 自增 | 教师用户登录名(教师用户表的主键,不为空) |
| u\_password | varchar | 教师登录密码(不为空) |
| u\_name | varchar | 教师真实名称 |
| u\_age | varchar | 教师年龄(不为空) |
| u\_phone | varchar | 教师联系方式(不为空) |
| u\_post | varchar | 教师职位(不为空) |

**2.权限表**

权限表(Auth\_tb)用与存放系统用户的权限信息,如下图4-3-2.2所示

|  |  |  |
| --- | --- | --- |
| 字段名 | 字段类型 | 字段描述及说明 |
| a\_id | Int 自增 | 权限ID(权限表的主键,不为空) |
| a\_name | varchar | 权限名称(不为空) |
| a\_explain | varchar | 权限描述(不为空) |

**3.管理员信息表**管理员信息表(Role\_tb)用户存放管理员的基本信息,如下图4-3-2.3所示

|  |  |  |
| --- | --- | --- |
| 字段名 | 字段类型 | 字段描述及说明 |
| r\_id | Int 自增 | 管路员登录名(管理员信息表的主键,不为空) |
| u\_password | varchar | 管理员登录密码(不为空) |
| r\_name | varchar | 管理员登录真实姓名(不为空) |
| r\_age | varchar | 教师年龄(不为空) |
| r\_phone | varchar | 教师联系方式(不为空) |
| r\_post | varchar | 教师职位(不为空) |

第5章 系统实现与测试

## 5.1 通用模块

### 5.1.1 工具类

### 5.1.2 数据源配置

1. 数据源配置
2. MyBatis相关配置
3. 事务管理器配置

## 5.2 功能模块

## 5.3系统运行及测试

测试贯穿于系统的整个开发过程，这里仅列出主要的界面测试用例。

### 5.3.1 登录测试用例

### 5.3.2 添加用户测试用例

### 5.3.3 修改用户测试用例

### 5.3.4 查询用户测试用例

### 5.3.5 删除用户测试用例

第6章 结 束 语

致谢

# 